Neural Language Generation for Spoken Dialogue Systems

Tsung-Hsien (Shawn) Wen and Steve Young
Outline

- Intro
- RNN Generator
- Semantically Conditioned LSTM
- Experiments
- Adaptation – A preliminary work
- Conclusion
Outline

- Intro
- RNN Generator
- Semantically Conditioned LSTM
- Experiments
- Adaptation – A preliminary work
- Conclusion
Spoken Dialogue System

Diagram:
- Speech Recognition
- Language Understanding
- Dialogue Manager
- Language Generation
- Knowledge Base
- Web
NLG: Problem Definition

- Given a meaning representation, map it into natural language utterances.

**Dialogue Act**

| Inform(restaurant=Seven_days, food=Chinese) |

**Realisations**

- Seven days is a restaurant serving Chinese.
- Seven days is a Chinese restaurant.

- What do we care about?
  - adequacy, fluency, readability, variation
    (Stent et al 2005)
Traditional pipeline approach

Inform( name=Z_House, price=cheap )

Z House is a cheap restaurant.

Dialoge Act

Tree-like template

Utterance
Problems

- Scalability
  - Grammars are handcrafted.
  - Require expert knowledge.

\[
\begin{align*}
A & \rightarrow \text{mm}, \text{Pr}(0.11) \mid \text{mh}, \text{Pr}(0.67) \mid \text{hh}, \text{Pr}(0.22) \\
B & \rightarrow \text{mm}, \text{Pr}(0.68) \mid \text{hm}, \text{Pr}(0.23) \mid \text{hh}, \text{Pr}(0.09) \\
C & \rightarrow \text{mm}, \text{Pr}(0.58) \mid \text{hm}, \text{Pr}(0.42) \\
T & \rightarrow \text{hQA}, \text{Pr}(0.12) \mid \text{hQB}, \text{Pr}(0.18) \mid \text{APm}, \text{Pr}(0.16) \\
U & \rightarrow \text{ARC}, \text{Pr}(0.13) \mid \text{BB}, \text{Pr}(0.39) \mid \text{hOm}, \text{Pr}(0.15) \\
V & \rightarrow \text{ARA}, \text{Pr}(0.16) \mid \text{AB}, \text{Pr}(0.66) \mid \text{BC}, \text{Pr}(0.08) \\
W & \rightarrow \text{BRA}, \text{Pr}(0.10) \mid \text{CA}, \text{Pr}(0.08) \mid \text{CR}, \text{Pr}(0.07) \\
R & \rightarrow \text{tWm}, \text{Pr}(0.14) \mid \text{mWm}, \text{Pr}(0.22) \mid \text{mWh}, \text{Pr}(0.23) \\
Q & \rightarrow \text{AVh}, \text{Pr}(0.28) \mid \text{BVm}, \text{Pr}(0.55) \mid \text{BVh}, \text{Pr}(0.06) \\
P & \rightarrow \text{ILB}, \text{Pr}(0.14) \mid \text{mUC}, \text{Pr}(0.22) \mid \text{hLIA}, \text{Pr}(0.20) \\
O & \rightarrow \text{ATA}, \text{Pr}(0.86) \mid \text{CTC}, \text{Pr}(0.14) \\
X & \rightarrow \text{X}, \text{Pr}(0.35) \mid \epsilon, \text{Pr}(0.55) \\
S & \rightarrow \text{[TX]}, \text{Pr}(1.00)
\end{align*}
\]
Problems

- Boring
  - Frequent repetition of outputs.
  - Non-colloquial, awkward utterances.

*Seven Days is a nice restaurant in the expensive price range, in the north part of the town, if you don’t care about what food they serve.*
Outline

- Intro
- **RNN Generator**
- Semantically Conditioned LSTM
- Experiments
- Adaptation – A preliminary work
- Conclusion
Recurrent Generation Model

Inform(name=Seven_Days, food=Chinese)

\[
\begin{bmatrix}
0, 0, 1, 0, 0, ..., 1, 0, 0, ..., 1, 0, 0, 0, 0, 0, ...
\end{bmatrix}
\]

dialog act 1-hot representation

\[
\text{delexicalisation}
\]

RNNLM (Mikolov et al 2010)
Recurrent Generation Model

- Gates are controlled by **exact matching** of features and generated tokens.
- Apply a decay factor $\delta < 1$ on feature values.
- Binary slots/special values need to be additionally handled.

SLOT_NAME serves SLOT_FOOD.

1 1

NAME FOOD
Over-generation & Reranking

- Generate a bunch of candidate utterances.
- Rerank them!

  Seven days is a good restaurant in the south. \(0.9\)
  There is no restaurant in the south. \(0.2\)
  Seven days is in the south part of town. \(0.7\)

- Simple & variation included.

(Oh & Rudnicky 2000)
CNN Semantic Reranker

Target dialogue act: inform(name=Seven_days, food=Chinese)  
Generated candidate: </s> SLOT_NAME serves SLOT_FOOD. </s>  

(Kalchbrenner et al., 2014)
Backward Reranker

- Train a RNN with utterances reversed.
  - In order to consider backward context
  - Ex. “Seven Days is an exceptional restaurant.”

- Reranking Score:
  - $LL_{ForwardRNN} + LL_{BackwardRNN} - LOSS_{CNN}$
Generated Examples

\begin{tabular}{|l|}
\hline
\texttt{inform_only(name='red victorian bed breakfast',accepts_credit_cards='yes',near='haight',has_internet='yes')} \\
\hline
red victorian bed breakfast is the only hotel near haight and accepts credit cards and has internet . \\
red victorian bed breakfast is the only hotel near haight and has internet and accepts credit cards . \\
red victorian bed breakfast is the only hotel near haight that accept credit cards and offers internet . \\
the red victorian bed breakfast has internet and near haight , it does accept credit cards . \\
the red victorian bed breakfast is the only hotel near haight that accepts credit cards , and offers internet . \\
\hline
\end{tabular}
Generated Examples

inform_no_match(area=tenderloin)

there are no restaurants in the tenderloin area.

there are 0 restaurants in the tenderloin area.

unfortunately there are 0 restaurants in the tenderloin area.

i could not find any restaurants in tenderloin.

there are no restaurants in tenderloin.
Generated Examples

?request(food)
what type of food are you looking for ?
what type of food would you like ?
what kind of food are you looking for ?
what type of food are you interested in ?
what kind of food would you like ?
Outline

- Intro
- RNN Generator
- **Semantically Conditioned LSTM**
- Experiments
- Adaptation – A preliminary work
- Conclusion
SC-LSTM

- **Original LSTM cell**
  \[\begin{align*}
  i_t &= \sigma(W_{wi}w_t + W_{hi}h_{t-1}) \\
  f_t &= \sigma(W_{wf}w_t + W_{hf}h_{t-1}) \\
  o_t &= \sigma(W_{wo}w_t + W_{ho}h_{t-1}) \\
  \hat{c}_t &= \tanh(W_{wc}w_t + W_{hc}h_{t-1}) \\
  c_t &= f_t \odot c_{t-1} + i_t \odot \hat{c}_t \\
  h_t &= o_t \odot \tanh(c_t)
  \end{align*}\]

- **DA cell**
  \[\begin{align*}
  r_t &= \sigma(W_{wr}w_t + W_{hr}h_{t-1}) \\
  d_t &= r_t \odot d_{t-1}
  \end{align*}\]

- **Modify \(C_t\)**
  \[c_t = f_t \odot c_{t-1} + i_t \odot \hat{c}_t + \tanh(W_{dc}d_t)\]

**Diagram**

A diagram illustrating the flow of information through the LSTM and DA cells, including the equations for input, forget, and output gates, as well as the cell state and hidden state equations. The diagram shows the interaction between the LSTM cell and the DA cell, and how they process input sequences.

**Example**

Inform(name=Seven_Days, food=Chinese)

**Dialog Act 1-Hot Representation**

Note: The diagram and equations are adapted from Hochreiter and Schmidhuber (1997).
Visualization
SC-LSTM

- Cost function
  \[ F(\theta) = \sum_t p_t^T \log(y_t) + \|d_T\| + \sum_{t=0}^{T-1} \eta \xi \|d_{t+1} - d_t\| \]

- 1st term: Log-likelihood
- 2nd term: make sure rendering all the information needed
- 3rd term: close only one gate each time step.

(Hochreiter and Schmidhuber, 1997)
Intuition behind the 3\textsuperscript{rd} term

\[ \eta = 0.01, \xi = 100 \]
Deep Architecture
Deep Architecture

- Techniques applied
  - Skip connection
    (Graves et al 2013)
  - RNN dropout
    (Srivastava et al 2014)
Outline

- Intro
- RNN Generator
- Semantically Conditioned LSTM
- Experiments
- Adaptation – A preliminary work
- Conclusion
Setup

- Data collection:
  - SFX restaurant/hotel domains
## Ontologies

<table>
<thead>
<tr>
<th></th>
<th>SF Restaurant</th>
<th>SF Hotel</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>act type</strong></td>
<td>inform, inform only, reject, confirm, select, request, reqmore, goodbye</td>
<td></td>
</tr>
<tr>
<td><strong>shared</strong></td>
<td>name, type, *pricerange, price, phone, address, postcode, *area, *near</td>
<td></td>
</tr>
</tbody>
</table>

**bold**=binary slots, *=slots can take “don’t care” value
Setup

- Data collection:
  - SFX restaurant/hotel domains
  - Workers recruited from Amazon MT.
  - Asked to generate system responses given a DA.
  - Result in ~5.1K utterances, 228/164 distinct acts.

- Training: BPTT, L2 reg, SGD w/ early stopping.
  - train/valid/test: 3/1/1, data up-sampling

Available at: https://www.repository.cam.ac.uk/handle/1810/251304
Corpus-based Evaluation

- Test set: ~1K utterances each domain
- Metrics: BLEU-4 (against multiple references), ERR(slot error rates)
- Averaged over 5 random initialised networks.
- Over-gen 20, evaluate on top-5
- Models compared:
  - handcrafted generator (hdc)
  - kNN example-based generator (kNN)
  - class-based LM generator (classlm, O&R 2000)
  - rnn-based generator (rnn, Wen et al 2015)
Corpus-based Evaluation

Selection scheme: 5/20

Model

<table>
<thead>
<tr>
<th></th>
<th>hdc</th>
<th>knn</th>
<th>classlm</th>
<th>rnn</th>
<th>sc-lstm</th>
<th>+deep</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLEU</td>
<td>0.45</td>
<td>0.6</td>
<td>0.7</td>
<td>0.75</td>
<td>0.8</td>
<td>0.85</td>
</tr>
</tbody>
</table>

Restaurant

Hotel
Corpus-based Evaluation

Selection scheme: 5/20

- hdc
- knn
- classlm
- rnn
- sc-lstm
- +deep

Model

Restaurant
Hotel

ERR (%)
Human Evaluation

- **Setup**
  - Judges (~60) recruited from Amazon MT.
  - Asked to evaluate two system responses pairwise.
  - Comparing classlm, rnn, sc-lstm, and +deep

- **Metrics:**
  - Informativeness, Naturalness (rating out of 3)
  - Preference
# Human Evaluation

<table>
<thead>
<tr>
<th>Method</th>
<th>Informativeness</th>
<th>Naturalness</th>
</tr>
</thead>
<tbody>
<tr>
<td>+deep</td>
<td>2.58</td>
<td>2.51</td>
</tr>
<tr>
<td>sc-lstm</td>
<td>2.59</td>
<td>2.50</td>
</tr>
<tr>
<td>rnn</td>
<td>2.53</td>
<td>2.42*</td>
</tr>
<tr>
<td>classlm</td>
<td>2.46**</td>
<td>2.45</td>
</tr>
</tbody>
</table>

* $p < 0.05$  ** $p < 0.005$
## Human Evaluation

<table>
<thead>
<tr>
<th>Pref. %</th>
<th>classlm</th>
<th>rnn</th>
<th>sc-lstm</th>
<th>+deep</th>
</tr>
</thead>
<tbody>
<tr>
<td>classlm</td>
<td>-</td>
<td>46.0</td>
<td>40.9**</td>
<td>37.7**</td>
</tr>
<tr>
<td>rnn</td>
<td>54.0</td>
<td>-</td>
<td>43.0</td>
<td>35.7*</td>
</tr>
<tr>
<td>sc-lstm</td>
<td>59.1*</td>
<td>57</td>
<td>-</td>
<td>47.6</td>
</tr>
<tr>
<td>+deep</td>
<td>62.3**</td>
<td>64.3**</td>
<td>52.4</td>
<td>-</td>
</tr>
</tbody>
</table>

*p < 0.05 **p < 0.005
Outline

- Intro
- RNN Generator
- Semantically Conditioned LSTM
- Experiments
  - Adaptation – A preliminary work
- Conclusion
Attentive Encoder-Decoder

- **Embedding**
  \[ z_i = s_i + v_i \]

- **Attention**
  \[ \beta_{t,i} = q^T \tanh(W_{hm} h_{t-1} + W_{mm} z_i) \]
  \[ \omega_{t,i} = e^{\beta_{t,i}} / \sum_i e^{\beta_{t,i}} \]
  \[ d_t = a \oplus \sum_i \omega_{t,i} z_i \]

- **Generation**
  - Typical LSTM

(Mei et al 2015)
Experiments

- On new laptop ontology

<table>
<thead>
<tr>
<th>act type</th>
<th>inform, inform_only_match, inform_no_match, inform_count, inform_all, inform_no_info, recommend, compare, confirm, select, suggest, request, request_more, goodbye</th>
</tr>
</thead>
<tbody>
<tr>
<td>slots</td>
<td>family*, battery_rating*, drive_range*, is_for_business*, price_range*, weight_range*, warranty, battery, design, dimension, utility, weight, platform, memory, price, drive, processor</td>
</tr>
</tbody>
</table>

| **bold**=binary slots, *=slots can take don’t care value |

- Comparing performance and adaptation capability with SC-LSTM.
From scratch
Adapt from Rest+Hotel to Laptop
Outline

- Intro
- RNN Generator
- Semantically Conditioned LSTM
- Experiments
- Adaptation – A preliminary work
- Conclusion
Conclusion

- NLG can be learned N2N from data.
- Learn LM & slot gating control signal jointly
- Corpus-based/Human evaluation.
- More colloquial, more scalable.
- Potential for open domain SDS.
Papers


Thank you! Questions?

Tsung-Hsien Wen is supported by a studentship funded by Toshiba Research Europe Ltd, Cambridge Research Laboratory